

Coroutines mean that the thing that looks like a stack variable may not technically be one

 devblogs.microsoft.com/oldnewthing/20171211-00

December 11, 2017



Raymond Chen

Some time ago, I pointed out the `GetCurrentThreadStackLimits` function which lets you determine whether a pointer points into the stack.

Note, however, that if the local variable is captured by a coroutine, The introduction of coroutines into the C++ language means that what looks like a local variable may end up being hoisted into a heap-allocated object representing the coroutine itself (like, say, a `std::future`).

Something to bear in mind if you're going to be making decisions based on whether a variable is on the stack.

[Raymond Chen](#)

Follow

